

[illegible]

3

Sy

MT

MT

MT

MT
MT

MT
MT

MT
MT

MT
MTMT
MT

MT

MT

MT

MT

MT
MT

MT
MT

MT
MTMT
MT

MT

MT

MT

MI

MT
MT

MT
MTMT
MT

MT

M1
M2

W1
W1
W1

41
 42

M1

1

1

1

1

1

—

```

LL          IIIII
LL          IIIII
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LL          II
LLLLLLLLLL IIIII
LLLLLLLLLL IIIII
SSSSSSSSS
SSSSSSSSS
SS
SS
SS
SS
SSSSSS
SSSSSS
SS
SS
SS
SS
SSSSSSSSS
SSSSSSSSS

```

(2) 45
(3) 75

DECLARATIONS
OTSSPOWLULU - LU ** LU Exponentiation Routine

```
0000 1 .TITLE OTSS$POWLULU - LU ** LU Exponentiation Routine
0000 2 .IDENT /1-001/ ; File: OTSPOWLUL.MAR, Edit: SBL1001
0000 3
0000 4 *****
0000 5 *
0000 6 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
0000 7 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0000 8 * ALL RIGHTS RESERVED. *
0000 9 *
0000 10 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000 11 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000 12 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000 13 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000 14 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000 15 * TRANSFERRED. *
0000 16 *
0000 17 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000 18 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000 19 * CORPORATION. *
0000 20 *
0000 21 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000 22 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0000 23 *
0000 24 *
0000 25 *****
0000 26
0000 27 ++
0000 28 FACILITY: Mathematics Library
0000 29
0000 30 ABSTRACT:
0000 31
0000 32 This module contains OTSS$POWLULU which takes an unsigned longword base
0000 33 to an unsigned longword power resulting in an unsigned longword.
0000 34
0000 35 ENVIRONMENT: Runs at any access mode, AST Reentrant
0000 36
0000 37
0000 38 AUTHOR: Steven B. Lionel, CREATION DATE: 2-September-1980
0000 39
0000 40 MODIFIED BY:
0000 41
0000 42 1-001 - Original. SBL 2-September-1980
0000 43 --
```



```

0000 45      .SBTTL  DECLARATIONS
0000 46      :
0000 47      : LIBRARY MACRO CALLS:
0000 48      :
0000 49      :     NONE
0000 50      :
0000 51      : EXTERNAL DECLARATIONS:
0000 52      :
0000 53      :     .DSABL  GBL           : Force all external symbols to be declared
0000 54      :     .EXTRN  MTH$$SIGNAL : Signal a math error
0000 55      :     .EXTRN  MTH$K_UNDEXP : Undefined exponentiation error code
0000 56      :
0000 57      : MACROS:
0000 58      :
0000 59      :     NONE
0000 60      :
0000 61      : EQUATED SYMBOLS:
0000 62      :
0000 63      :     NONE
0000 64      :
0000 65      : OWN STORAGE:
0000 66      :
0000 67      :     NONE
0000 68      :
0000 69      : PSECT DECLARATIONS:
0000 70      :
0000 71      : .PSECT _MTH$CODE PIC, USR, CON, REL, LCL, SHR, -
0000 72      :     EXE, RD, NOWRT, LONG
0000 73

```

```
0000 75 .SBTTL OTSS$POWLULU - LU ** LU Exponentiation Routine
0000 76 :++
0000 77 : FUNCTIONAL DESCRIPTION:
0000 78 :
0000 79 : This routine takes an unsigned longword integer base to an unsigned
0000 80 : longword integer power yielding an unsigned longword integer result.
0000 81 :
0000 82 : Overflow can not happen in this routine. If the result or intermediate
0000 83 : result is greater than 32 bits, the low 32 bits are used.
0000 84 :
0000 85 : CALLING SEQUENCE:
0000 86 :
0000 87 : result.wlu.v = OTSS$POWLULU (base.rlu.v, exponent.rlu.v)
0000 88 :
0000 89 : FORMAL PARAMETERS:
0000 90 :
00000004 0000 91 : base = 4 ; Unsigned longword integer base by value
00000008 0000 92 : exponent = 8 ; Unsigned longword integer exponent by value
0000 93 :
0000 94 : IMPLICIT INPUTS:
0000 95 :
0000 96 : NONE
0000 97 :
0000 98 : IMPLICIT OUTPUTS:
0000 99 :
0000 100 : NONE
0000 101 :
0000 102 : FUNCTION VALUE:
0000 103 :
0000 104 : The unsigned longword integer result is returned in R0.
0000 105 :
0000 106 : SIDE EFFECTS:
0000 107 :
0000 108 : MTH$_UNDEXP if both the base and exponent are zero
0000 109 :
0000 110 :--
0000 111 :
0004 0000 112 .ENTRY OTSS$POWLULU, ^M<R2> ; IV must be disabled
0002 113 :
50 01 D0 0002 114 MOVL #1, R0 ; Intialize result
52 04 AC D0 0005 115 MOVL base(AP), R2 ; Get base
51 08 AC D0 0009 116 MOVL exponent(AP), R1 ; Get exponent
1B 13 000D 117 BEQL ZERO_EXP ; If exponent is zero, result is 1
000F 118 : unless base is zero.
000F 119 :
000F 120 :+
000F 121 : Scan exponent bits from right to left, squaring base each time through loop.
000F 122 : If an exponent bit is set, multiply the current result by the current base.
000F 123 :--
000F 124 :
000F 125 LOOP:
13 51 E8 000F 126 BLBS R1, PARTIAL ; If exponent bit set, get partial result
0012 127 LOOP1:
51 51 FF 8F 78 0012 128 ASHL #-1, R1, R1 ; Drop rightmost exponent bit
51 80000000 8F CA 0017 129 BICL2 #^X80000000, R1 ; Clear leftmost bit
19 13 001E 130 BEQL EXIT ; If zero, we're done
52 52 C4 0020 131 MULL2 R2, R2 ; Square current base
```

OTSS\$POWLULU
1-001

N 6

- LU ** LU Exponentiation Routine 16-SEP-1984 02:03:02 VAX/VMS Macro V04-00 Page 4
OTSS\$POWLULU - LU ** LU Exponentiation Ro 6-SEP-1984 11:28:42 [MTHRTL.SRC]OTSPOWLUL.MAR;1 (3)

```

EA 11 0023 132 ; If overflow, use low 32 bits
      0023 133 ; Continue searching
      0025 134 BRB LOOP
      0025 135 PARTIAL:
50 52 C4 0025 136 MULL2 R2, R0 ; Get new partial result
      0028 137 ; If we overflow, use low 32 bits
E8 11 0028 138 BRB LOOP1 ; Continue searching
      002A 139
      002A 140 ZERO_EXP:
52 D5 002A 141 TSTL R2 ; Exponent is zero, is base?
      0B 12 002C 142 BNEQ EXIT ; If not, result is 1
7E 00'8F 9A 002E 143 MOVZBL #MTH$K UNDEXP, -(SP) ; If so, it's an error
00000000'GF 01 FB 0032 144 CALLS #1, G^MTH$$SIGNAL ; Signal the error
      0039 145
      0039 146 EXIT: RET ; Return with result in R0
      003A 147
      003A 148 .END ; End of module OTSS$POWLULU
```


OTSSPOWLULU
Symbol table

- LU ** LU Exponentiation Routine ^{B 7}

16-SEP-1984 02:03:02
6-SEP-1984 11:28:42

VAX/VMS Macro V04-00
[MTHRTL.SRC]OTSPOWLUL.MAR;1

Page 5
(3)

BASE	=	00000004		
EXIT		00000039	R	01
EXPONENT	=	00000008		
LOOP		0000000F	R	01
LOOP1		00000012	R	01
MTH\$SIGNAL		*****	X	00
MTH\$K UNDEXP		*****	X	00
OTSSPOWLULU		00000000	RG	01
PARTIAL		00000025	R	01
ZERO_EXP		0000002A	R	01

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes														
ABS	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE				
MTH\$CODE	0000003A (58.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG				

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.08	00:00:00.82
Command processing	119	00:00:00.50	00:00:02.65
Pass 1	69	00:00:00.42	00:00:03.50
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	40	00:00:00.28	00:00:01.87
Symbol table output	2	00:00:00.03	00:00:00.22
Psect synopsis output	3	00:00:00.03	00:00:00.04
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	264	00:00:01.34	00:00:09.11

The working set limit was 900 pages.
1694 bytes (4 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 10 non-local and 0 local symbols.
148 source lines were read in Pass 1, producing 11 object records in Pass 2.
0 pages of virtual memory were used to define 0 macros.

! Macro library statistics !

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS\$:OTSPOWLUL/OBJ=OBJ\$:OTSPOWLUL MSRC\$:OTSPOWLUL/UPDATE=(ENH\$:OTSPOWLUL)

0265 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY